Computational Fluid Dynamics

Dr. Krishna M. Singh
Department of Mechanical and Industrial Engineering
Indian Institute of Technology - Roorkee

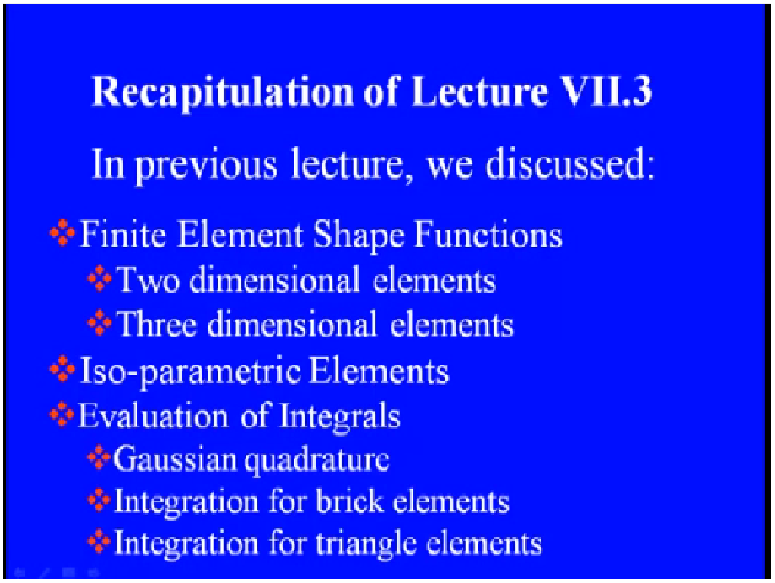
Lecture – 34
Application of FEM to Scalar Transport

Welcome to the last lecture in module 7 on finite element method. In the previous lectures, we have discussed the basic introduction to finite element method and we looked at specific formulations which we call weighted residual formulation. Then we had also had a brief look at the variation formulation and then we discussed few typical finite element shape functions, specifically the polynomial shape functions of Lagrange family and Serendipity family for rectangular elements.

We also discussed the shape functions for triangular and hexahedral elements and we also had a brief look at the numeric integration. Now in the last lecture, we are going to have a look at application of FEM to scalar transport; in particular, we will discuss the application of finite element in detail to 1-dimensional heat conduction problem.

So the aim of today's lecture will be to give you a detailed derivation of the method, how do we derive each of the element matrices, how do we complete the assembly process so that you can relate it and convert it into a computer program for numerical simulation. So let us have a recapitulation of what we did in lecture 3 of this module.

(Refer Slide Time: 01:53)

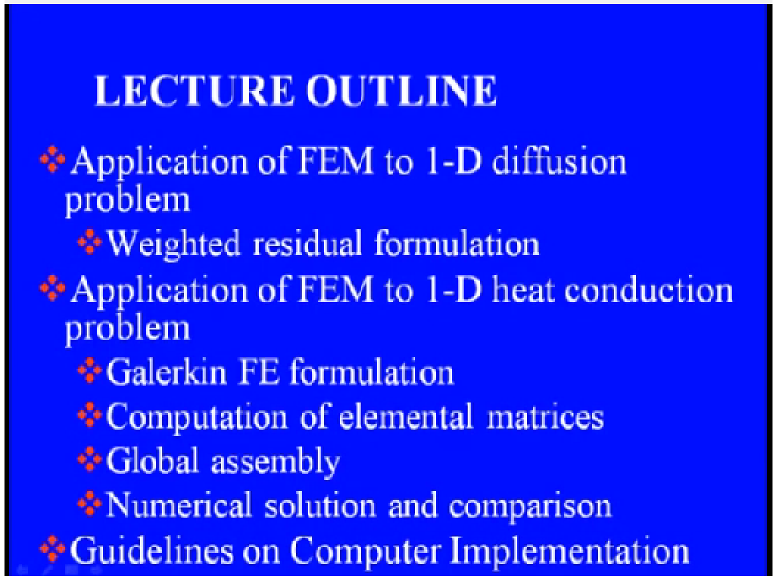


We discussed Finite Element Shape Functions for 2-dimensional elements and then 3-dimensional elements. We also discussed what we mean by Iso-parametric Elements which are used in the case of complex geometries where in we would use the same shape functions which we have used for variable variation to map a model the geometry of the element.

And then we discussed the evaluation of integrals, basically Gaussian Quadrature for the rectangular elements and we also mentioned that how do we apply simple 1-dimensional Gaussian Quadrature formula for the integration of brick elements in 3-dimensions and then we had a discussion on the integration for a triangular elements and we briefly mentioned special Quadrature formulae for triangular elements and just mentioned the possibility of a similar set of formulae for tetrahedral elements in 3-dimensions.

And today's lecture, we would apply finite element to the scalar transport problem since this is just 1 sample of allocation which will give you an idea that how do we proceed with a formulation of finite element method to a particular problem and in what way can that be translated into a computer program to solve a wide range of problems.

(Refer Slide Time: 03:24)



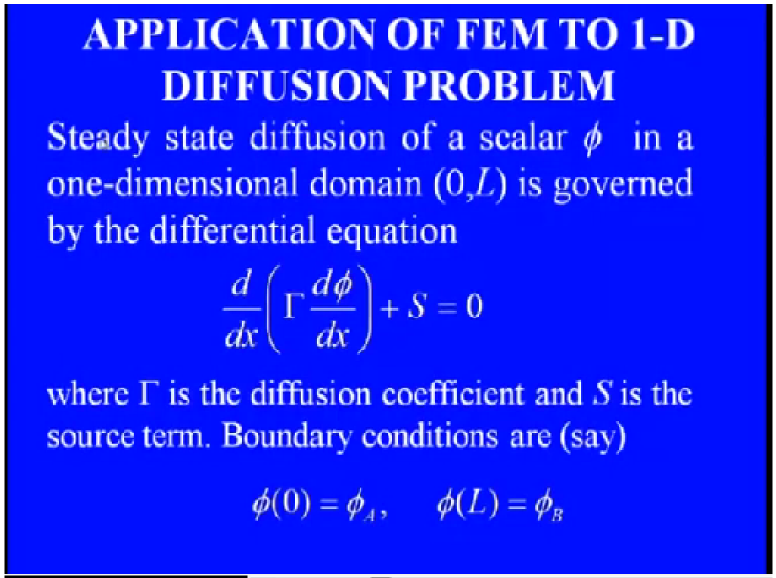
So we will first discuss application of FEM to 1-dimensional diffusion problem. We will have a look at weighted residual formulation. This is the formulating which is used most widely because of its wide applicability. We do not have to worry about existence of variational form or what we call a scalar functional which can be minimised to obtain a finite element formulation. So this weighted residual formulation can be applied to any given problem.

And then we will extend this application, rather we will specialise discussions on 1-dimensional diffusion problems to 1-dimensional heat conduction problem. We will have a detailed look at the Galerkin finite element formulation and then we will see how do we compute the elemental matrices. In this case, for 1-D, it would be possible for us to compute these matrices analytically and then we will perform the global assembly of elemental matrices.

And then we will discuss the numerical solution which we obtained from finite element and compare it with the solutions obtained for the same problem using finite difference method. In fact, we will take the same running example which we have taken earlier in our module 3 in finite difference and later on in the module on finite volume techniques and then we will discuss briefly the Guidelines of the Computer Implementation of finite element method which is most likely more complicated than finite volume or finite difference formulations.

So first let us have a look at our 1-dimensional problem.

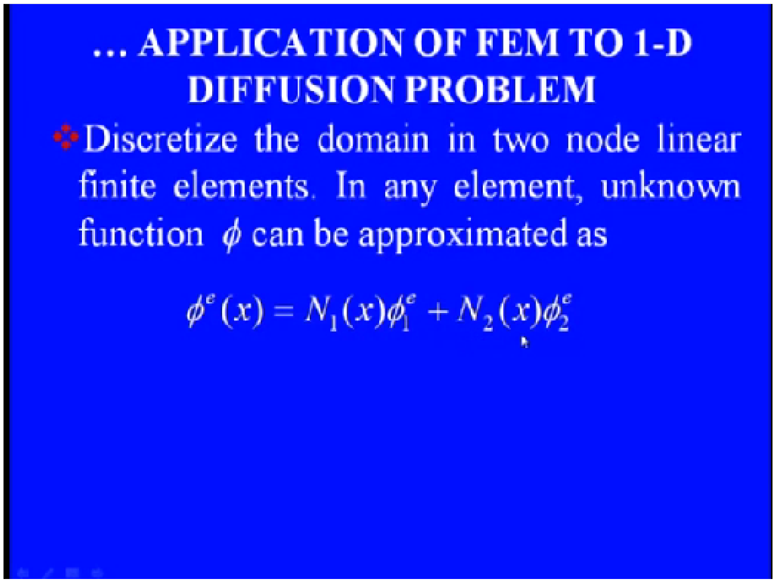
(Refer Slide Time: 05:17)



Let us have a look at the description of the problem and let us restrict ourselves to steady state situation. So if you got steady state diffusion of a scalar phi in 1-dimensional domain of length L, the extent of domain are from 0 to L, it is governed by the differential equation d/dx of gamma d phi/dx+S=0. Now here this gamma is diffusion coefficient and S is the source term which may or may not be dependent on this scalar phi.

It could be set of boundary condition, suppose for the time being that we assume Dirichlet boundary conditions at both ends that is the value of the phi is specified at both ends, 0 and L, so phi 0 is phi subscript A and phi at x=L=phi subscript B. The formulation would not depend on boundary condition per se. That is why we have just taken a set of typical boundary conditions. Now the first step of finite element formulation, we have now got to choose our discretization procedure.

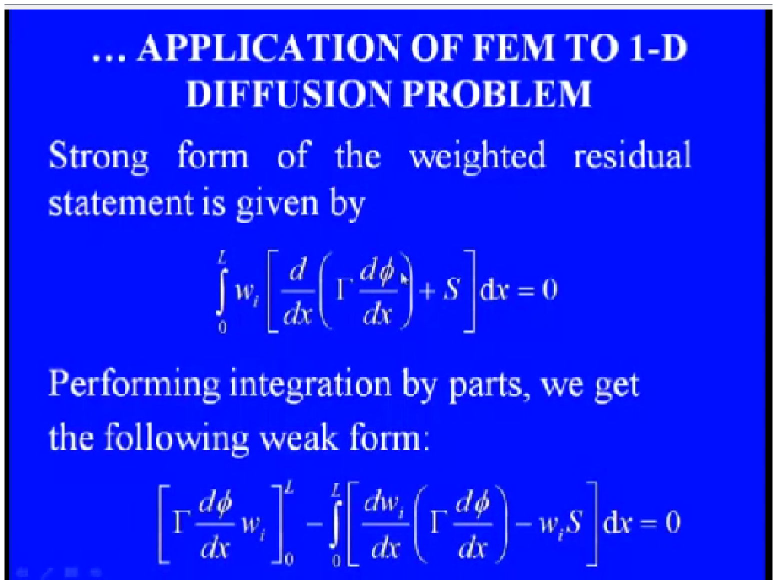
(Refer Slide Time: 06:26)



We have to choose which type of elements we want to use. So suppose just take the simplest one of them. Let us describe domain using 2 node linear finite elements. So in any element, the unknown function phi is approximated using this linear shape functions that phi ex=N1xphi 1e+N2x*phi 2e. Now here this N1x and N2x, they are what we call linear shape functions which depend on the spatial location x and phi 1e and phi 2e.

They are values of phi at local nodes 1 and 2 of the elements. These linear shape functions would satisfy the basic condition which we have discussed earlier that if they would have what we call Kronecker delta property of the nodes as well as they will satisfy the partition of unity. Now let us have a look at weighted residual formulation.

(Refer Slide Time: 07:33)



So suppose we have this particular case, we substitute for phi an approximate value or for the sake of simplicity, we have dropped that phi tilde, that we have dropped, so I have just written d phi/dx here. So d/dx gamma of d phi/dx+S, this gives us what we call our residual assuming this phi were an approximate solution, multiplied by a weight function, wi, integrate it over the domain that is 0 to L, dx=0.

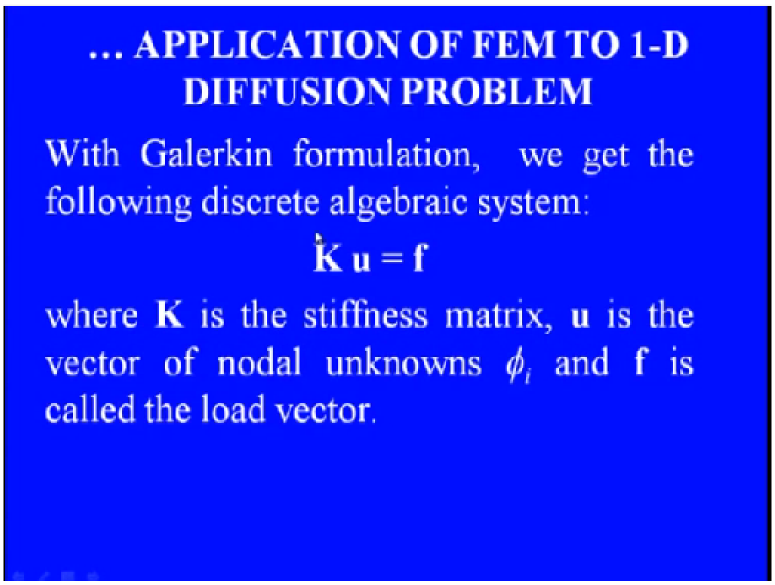
Now this is what is called a strong form because there are 2 derivatives here. So we will have to maintain at least a level of differentiability of phi that is we should be able to differentiate it twice. So that is a much stricter or stronger requirements on our shape functions which we would choose and the shape functions which I have chosen earlier, was it only linear elements, so the strong form will not work.

So what we need to do is, let us perform integration by parts to obtain the weak form. So integration by parts, we will focus primary in the first term which contains derivatives. So let us use the simple rule of calculus, so the first term when we integrate this omega i*d/dx gamma d phi/dx over 0 to L dx is, the first term would be wi*integral of the second term that is integral of d/dx of gamma d phi/dx, integral of the second term is simply gamma d phi/dx.

So that is why we got the first term as gamma d phi/dxwi at 2 limits of the domain, 0 to L, -, next term what we will get, the difference of the first term that was wi, so dwi/dx integral of the second term which was our this d/dx*gamma d phi/dx this integral is gamma d phi/dx, so that is why we get it here. So this is a second term which we get from the integration by parts of the second term involving second order derivative.

So -0 to L d of wi/dx gamma d phi/dx-wiS, that we have retained the second term, wi*source term as such and this whole thing has to be integrated from 0 to L, dx=0. Now this is a global weak form.

(Refer Slide Time: 10:19)

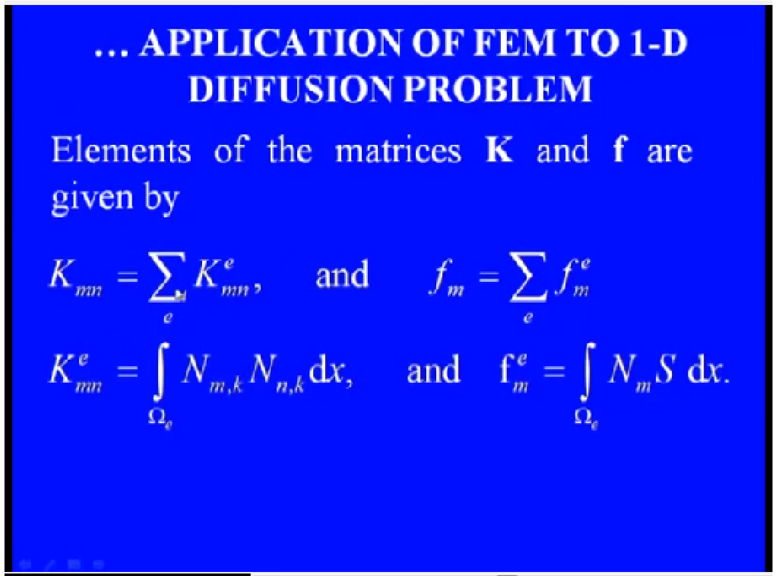


For finite element formulation, we will choose our appropriate local functions and in Galerkin formulation, what do we do, we will choose a weight function as a shape function for an element, so if you perform a substitution and express our integrals in terms of this shown over the elemental integrals, compute those integrals, sum it up, collect the complete system. So, then we would get the following discrete algebraic system, Ku=f.

Now this capital K is called a stiffness matrix for historical reasons, the finite element formulation which first applied in the structural mechanics and there the matrix which was obtained for this particular system that was referred to as stiffness because that was related to the stiffness of the structural elements. So same terminology is used in finite element application to any problem, so this matrix is historically called stiffness matrix.

U is the vector of a nodal unknowns phi i and we would follow the terminology which originates from structural applications finite element, this f is called the load vector and to obtain the solution, we have to solve this linear system to get our nodal unknowns.

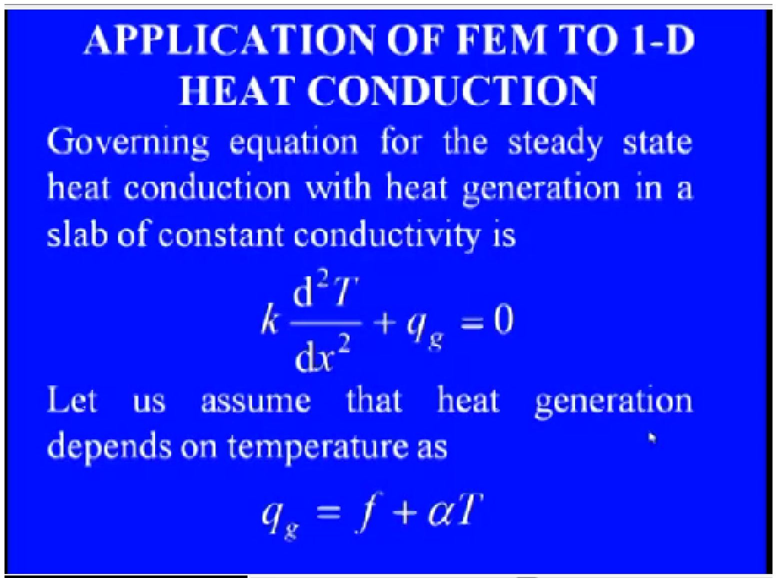
(Refer Slide Time: 11:42)



and matrices K and f, their elements were given by this Kmn=Sigma e Kmn for that particular element because we have taken the integral, we have summed it over the elemental integrals and so on and similarly, this fm is given by summing up all the terms corresponding to the load vector which we get from elements, so Sigma over e fme.

So now what are these Kmne, Kmne = integral of Sigma e that is integral over a given element which we have denoted here by Sigma e Nm, k, now here this presence of comma that indicates the derivative, similarly Nm, k*Nn, kdx and fme is integral over the element of NmSdx. Now there is some mysteries involved here, how do we get these terms, these are we are going to clarify when we take a special case of application finite element q 1-dimensional heat conduction problem.

(Refer Slide Time: 12:55)

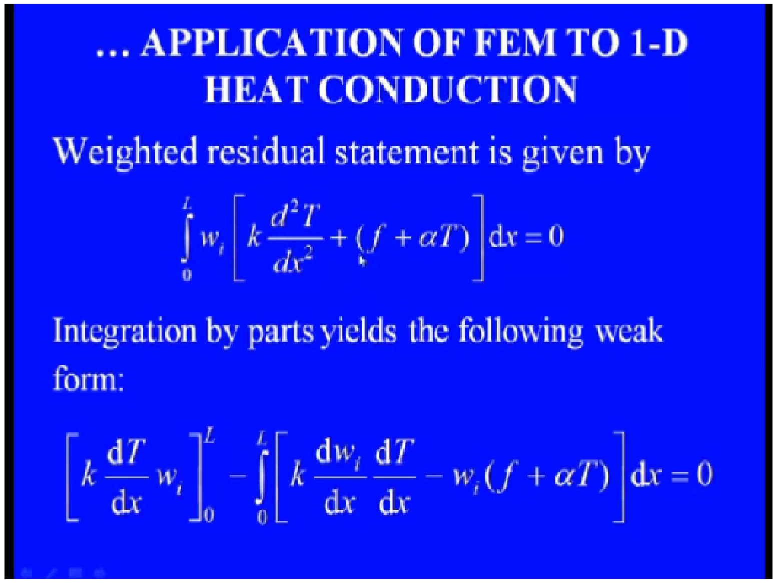


Here we will illustrate each step in detail (()) (12:58) in above. So now let us specialize our application that this is heat conduction equation specification of diffusion equation and in this situation, we have taken the conductivity to be constant. So let us take governing equation for steady state heat conduction with heat generation in a slab of constant conductivity.

So if you compare this scalar problems which we have discussed earlier, what we have done is, we have taken our diffusion coefficient to be constant which is conductivity in this case, so it becomes kd2T/dx square+qg=0. Now let us assume that heat generation depends on temperature, we have some situations where it would and majority of situations where it might simply be a constant.

So for this for generalised case, let us take qg = f+ this alpha*capital T, where this alpha is the coefficient of linear dependence and f is the constant part of our heat generation term.

(Refer Slide Time: 14:14)

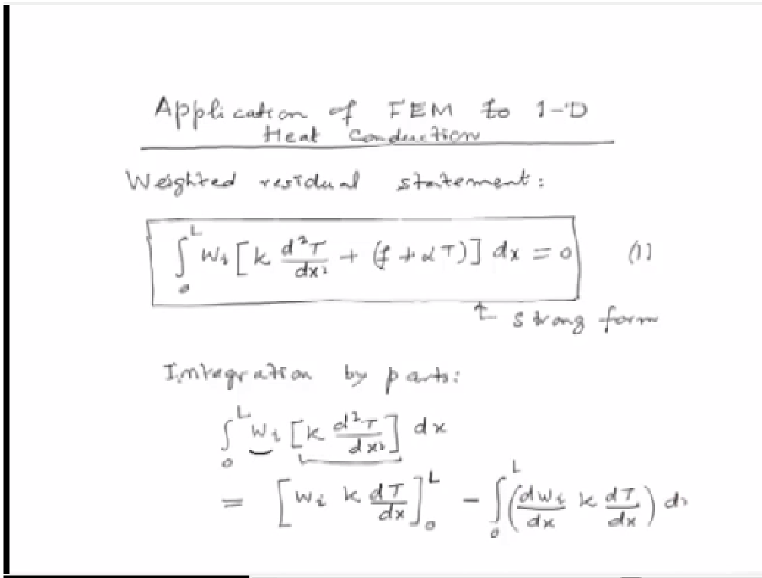


Now let us write our weighted residual statement where we will presume that we have substituted an approximate solution for capital T, so kd2T/dx square. For qg, we have substituted f+alpha T. So kd2T/dx square+(f+alpha delta T), this is what becomes our residual because we have presumed this capital T is our approximate solution to the exact function T.

So residual multiplied by a weight function wi integrate over the domain 0 to L dx, set it to 0, so this is our weighted residual statement in what we call strong form because it will presume that this approximate function or approximation of the temperature T must be differentiable at least twice. To reduce this differentiability requirement which will be transferred to the shape functions which are used to approximate T, let us perform integration by parts.

So once again let us do the integration of part only. We will focus on the first term which involves kd2T/dx square. The second term, we do not need to worry about. We will retain as such. So we are looking at the integral 0 to L of wi kd2t/dx square. Now we would like to put every step in detail. So now let us move on to our goal.

(Refer Slide Time: 15:30)

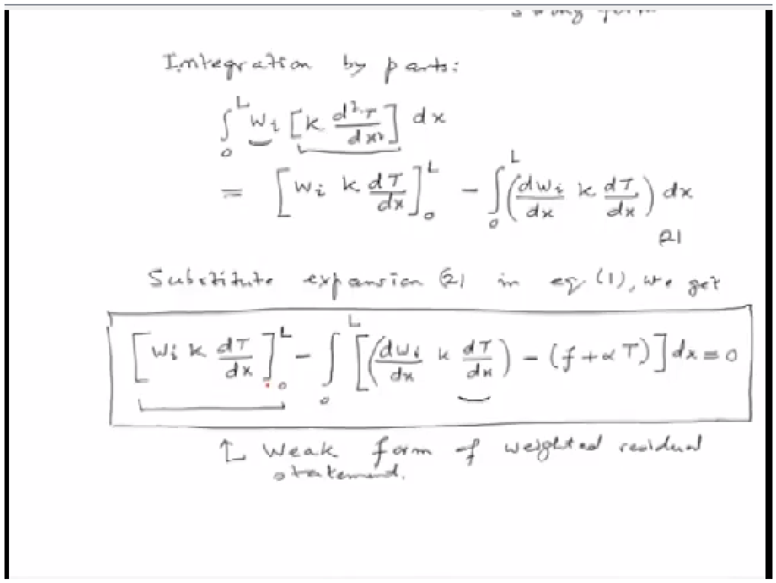


So application of FEM to 1-D heat conduction. So we will start of from our weighted residual statement which we have written earlier. So our weighted residual statement, that is the starting point of a finite element formulation. this is 0 to L wi[k*d2T/dx square+f+alpha T]dx=0 and this is what we said is our strong form. Let us call this equation 1. Now we would like to integrate the first term or our first part of this integral by parts, so let us try it out separately.

Integration by parts, so integral 0 to L wik*d2T/dx square dx. Now this consists of k is constants that we can take out of the integral. So our first term we will take as wi and second term for application of our rule of integration, we will take that as kd2t/dx square. So the rule of integration size retain the first term as such. Integration of second term that is k*d2T/dx square, that will give us k*dT/dx, different integrals we will have the contribution coming from both the ends -0 to L.

Differential of the first term, the first term was wi, so it will have dwi/dx k/dT/dx to dx.

(Refer Slide Time: 18:33)



So if you substitute this 2 in equation 1, so substitute expansion 2 in equation 1 and what do we get, the first term, wikdT/dx 0 to L-integral 0 to L, now we will take the second term on RHS of equation 2 and the remaining term which we had in equation 1 as such, so let us combine them together and write them in big brackets so we get [dwi/dx k*dT/dx- because we have taken minus on outside and first term what was positive, so this is negative here, f+alpha*T]dx=0.

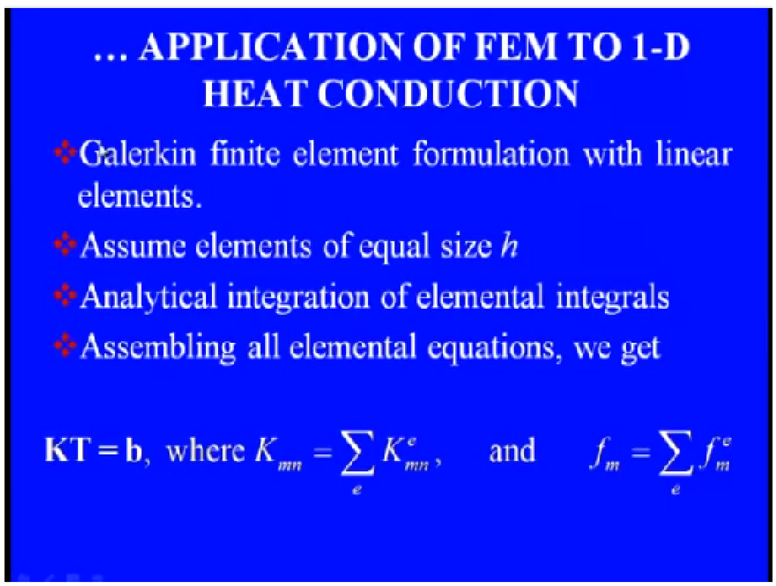
So this is what is our required weak form of this weighted residual statement because the part of the continuity requirement, now we have shifted from T to wi, shifting it from T to wi essentially means that if you are shifted from the interpolation functions on to the weight functions. So we have got much weaker continuity requirements on the shape functions which we are going to use to approximate T, so that is one aspect.

Second aspect which we can say is we look at the first term of this equation, wikdT/dx, now this has to be evaluated at the ends of the domain, that is boundary of the domain 0 to L, kdT/dx you can easily identify or link it with the flux term. So if their flux boundary condition is specified at the boundary, those are taken care of naturally here. So this is also one of the reasons why this flux specifications is referred to as the natural boundary condition which is incorporated as such exactly without any approximation in our finite element formulation.

So this is a strong contrast to what we have to do in finite difference formulation or in finite volume formulations where we have to make use of one-sided difference formula to take care of the flux specifications. So this in the essence say that we might have specific advantage because whenever there are natural boundary conditions specified that is flux boundary condition was specified, we would get a more accurate solution using finite element method compared to finite differences because the flux terms would be taken as such in their exact form.

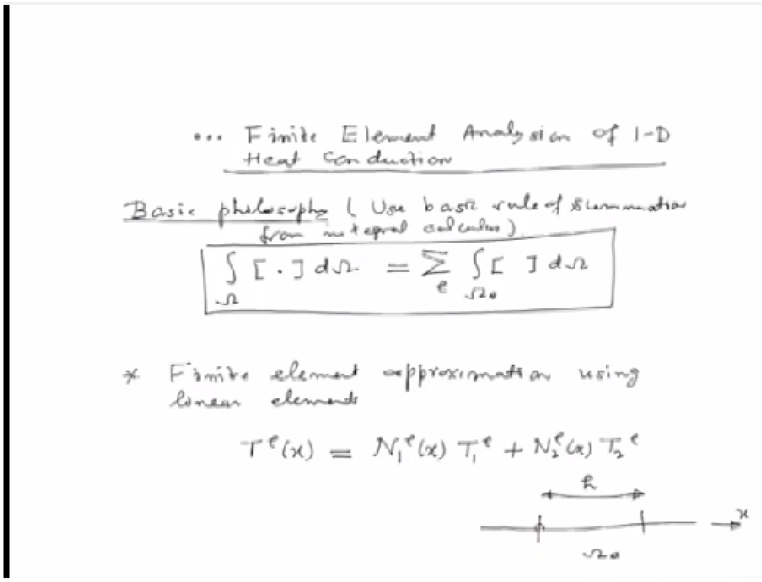
Okay, so that is what we have to get our, we have got our weak form and we have already mentioned this weak form incorporates our natural boundary condition or flux boundary condition exactly in our finite element formulation. Now, there are a few steps which we will take care. We will take them one by one, so what are the steps which we have got to perform.

(Refer Slide Time: 22:34)



The first we will have a look at what we call Galerkin finite element formulation and we will work with linear elements. We will assume our elements of equal sized h and then we will perform analytical integration of the elemental integrals. Thereafter we are going to perform elemental assembly. So now let us do each of these steps one by one.

(Refer Slide Time: 23:00)



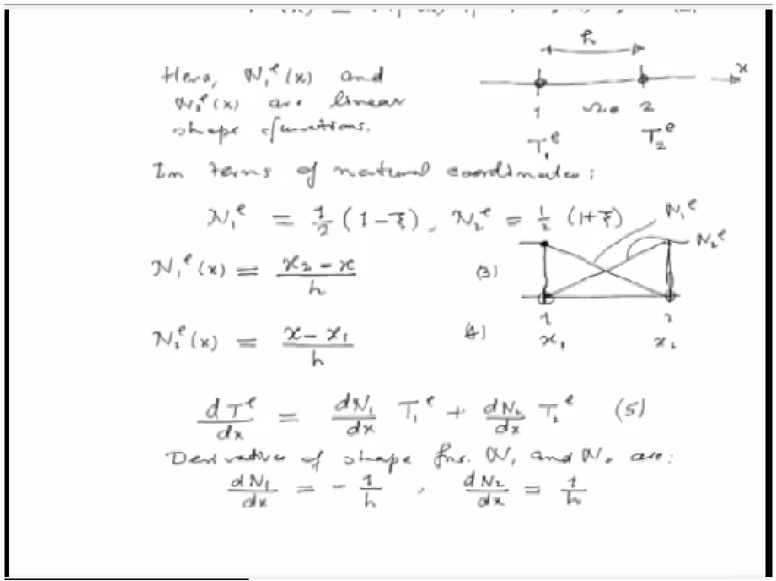
So we have already derived our global weighted residuals formulation and the integral which we had over entire domain, that can be summed up in terms of the integrals over each element. So that is to say that our basic philosophy here of finite element is simpler to this which we are going to make use of that integral over the complete domain of whatever terms we get, this can be represented as summation over e integral omega e of this function d omega, okay.

So this is the basic rule of calculus, so use basic summation rule of summation from integral calculus and let us also note down the first term which we had that was linked to the boundary nodes, so those will not be of any interest to us in evaluation or when we have considered each element separately. No let us have a look at the elemental integrals one by one, okay. So our elemental integrals before we go for that. We have to now use the approximation.

So finite element approximation using linear elements. So what we will do is, we would approximate the temperature in each element in terms of the linear shape functions N1ex*T1e+N2ex*T2e. So we are dealing with a particular element, let us call this as omega e, the extent of this since we are dealing 1-dimensional problem. Suppose this size is h or delta x whichever symbol you feel comfortable with, you can use it.

I would use h here to represent them as size which is node 1. This is node 2.

(Refer Slide Time: 26:20)



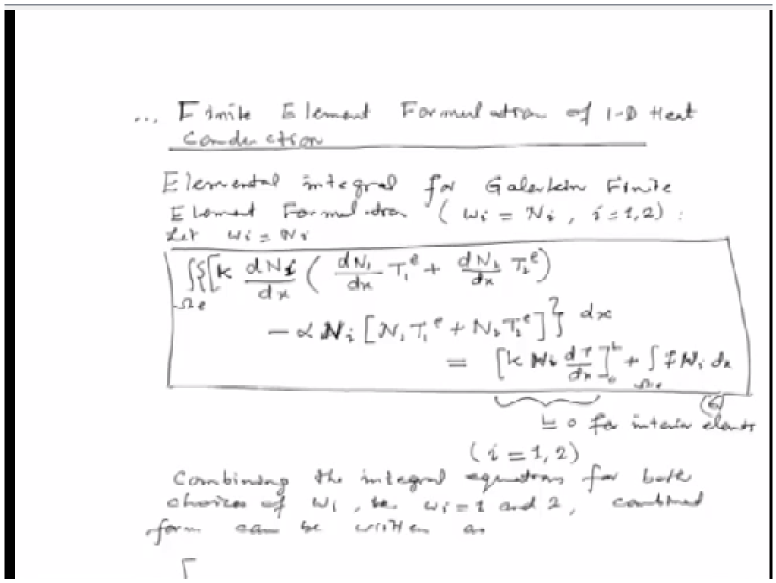
Value of the temperature at node 1, we will denote it by T1e, the value of the temperature at node 2 will be denoted by T2e. N1e and N2e are linear shape functions. So here N1ex and N2ex are linear shape functions. So if you want we can now use what we have learnt earlier in terms of the natural coordinates. So in terms of natural coordinates, what we have are N1e as given by 1/2×1-xi and N2 is given as 1/2*1+xi, okay.

Now here, it might be easier to work with the actual coordinates. So N1 and N2, they are both linear functions. So what we want to have, they would have linear variations, so their value reference and this is the plot for N1e, it will take a value of 1 at node 1 and its value should be 0 at node 2 that is the requirement what we Kronecker delta properties. Similarly, this is the plot of N2e which is 0 at node 1 and it takes value of 1 at node 2.

So can we write the expression directly in terms of x coordinates, x1 and x2. N1ex, this should be x2-x/h. Because when the value of x is x1, x2-x1 that is what gives us h. So N1 becomes 1 there, then x=x2, this x2-x2/h which is equal to 0. So this is how this is the expression for our shape function N1. Similarly expression for shape function N2, this would be x-x1/h. Now we know our shape functions.

What would be the derivatives because we require in our weighted residual formulation, dT/dx. So our dTe/dx, that is what we would need, dTe/dx, that would be dN1/dx*T1e+dN2/dx*T2e. So we need the derivative of N1 and N2 with respect to x. So these are straightforward as difference heat equations 3 and 4. So derivatives of shape function N1 and N2 are dN1/dx. You can clearly see this simply -1/h and dN2/dx, this would be simply 1/h.

(Refer Slide Time: 30:36)



Now let us have a look at the elemental integral for Galerkin finite element formulation. So you can make 2 choices that wi=Ni, i=1 to 2, this is what we do. In Galerkin finite element formulation, our weight function would be the same as the shape function. So elemental integral omega e, that constant terms which appeared in our weighted residual formulation, that was linked to the nodes on the boundaries. So let us for the time being discount those terms.

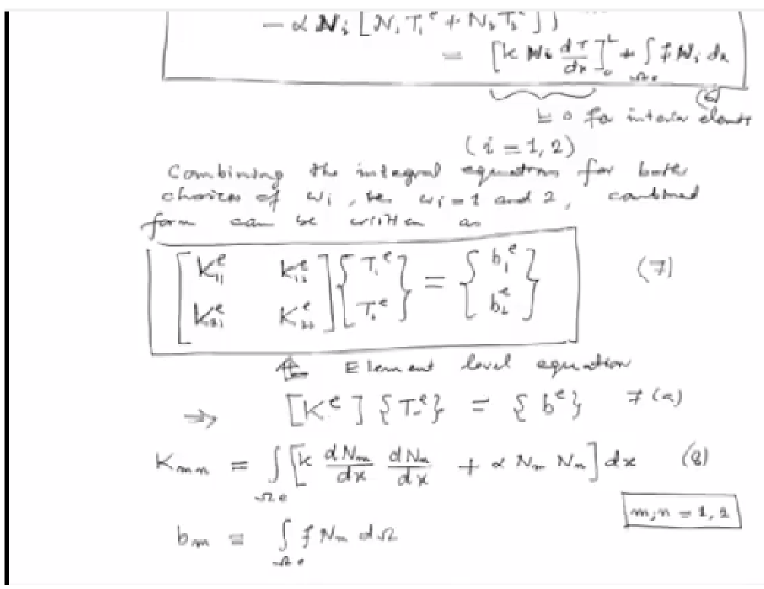
Let us only have a look at the integral over the domain which appears and now let us have a look at a typical term on our summation. So this is integral over our element omega e k*dNi/dx, let us take i=1, so we can take it as that is dN1/dx, N1 stands for our choice of the weight function, so let me choose that, this is wi=Ni. So dNi/dx within brackets, we want to find out the expansion for dT/dx, so dT/dx is dN1/dxT1e+dN2/dxT2e and the second term is -alpha*Ni, wi is now Ni.

We get for T, N1T1e+N2T2e, whole thing over dx and the constant part which we had which we have linked to the source terms, let us transfer it to the right-hand side which does not depend on temperature. So we will get k*widT/dx 0 to L +integral over omega e f*, this is Nidx. So this is our elemental integral equation. Where specific properties for these shape functions, so the N1 and N2, they are non-zero only over omega e. So this term is equal to 0 for interior elements, okay.

So we need to just worry about evaluating the remaining terms and its elemental integral. So now this equation, we have to write for 2 choices, for a linear element i could be 1 or 2. It is more convenient for us to write it in terms of matrix notation, that is we will get 1 equation we chose i=1 that is k*dN1/dx multiplied by this whole term, then alpha N2N1, this whole term, N1T1e+N2T2e and so on is equal to integral over omega efN1dx and we can write another equation when we take i=2, so we get kdN2/dx/dN1dxT1e+dN2dxT2e and so on.

Now we can combine these 2 equations and we can write 1 elemental equation in this form. So combining the integral equations for both choices of wi that is wi=1 and 2. Let us call the preceding equation as equation number 6. Combine form can be written as, now let us introduce the matrix notation here.

(Refer Slide Time: 37:22)



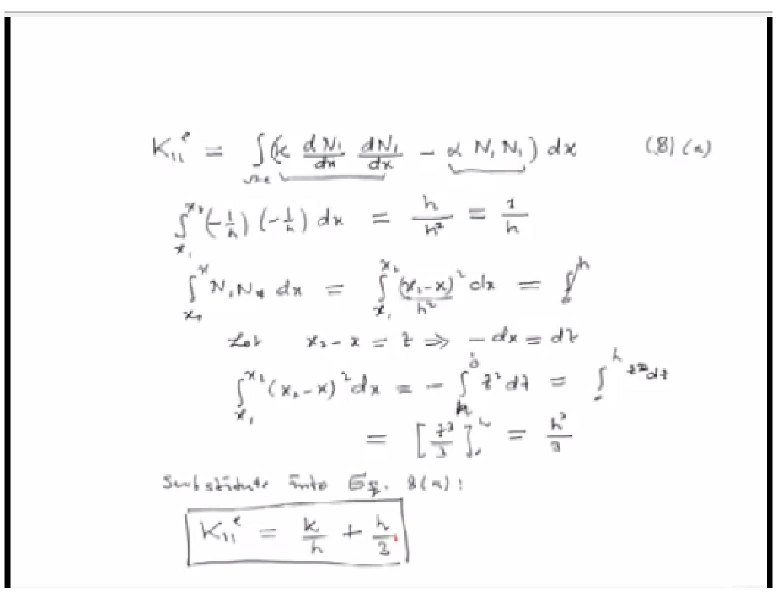
So we can have this k11, small k11e k12e small k21e, now let us, we can keep capital Ks here. Capital K22e*T1e T2e, this is equal to b1e and b2e, let us call this equation as 7. So whatever we have the terms which depend on T1 and T2, we have collected them on left-hand side and the right-hand side term has been evaluated separately and remember that one term which we left out kNidt/dx 0 to L that would appear or that would make a contribution only for the elements which are towards the ends, so that we can take care of separately.

Now what will these elemental terms look like. So this particular equation is also referred to as the element level equation, okay or in short-hand form matrix form, we can write it as Ke*vector Te=vector b3. Let us call this 7a. Now what will be the elements of this matrix, let us this try and workout this 7 is basically the collected version of equations 6 corresponding to 2 choices, i=1 and 2.

So basically what we will have is our Kmn, where m and n can take values from 1 and 2. This Kmn is given by the integral over omega e, the first index remember comes from our choice of w or weight function. So we will get this as k*dNm/dx. What else we will get, multiplied by dNn/dx, okay +the other contribution would come from our that alpha Ni terms. So alpha*Nmn. So you can verify that this is indeed the case by choosing values of m=1 and n and write down the expanded form for equation 6 for both choices of m and n. So let us call this as equation 8.

Similarly what do we get for bm, bm is nothing but integral of fNmd omega or dx for omega e and both these equations that we have chosen, m and n, they can take values from 1 to 2. Our next task is to obtain the explicit expressions for these terms K11 K223 or let us say our Kmn, so what will be Kmne. Let us have a look at K11e.

(Refer Slide Time: 41:56)

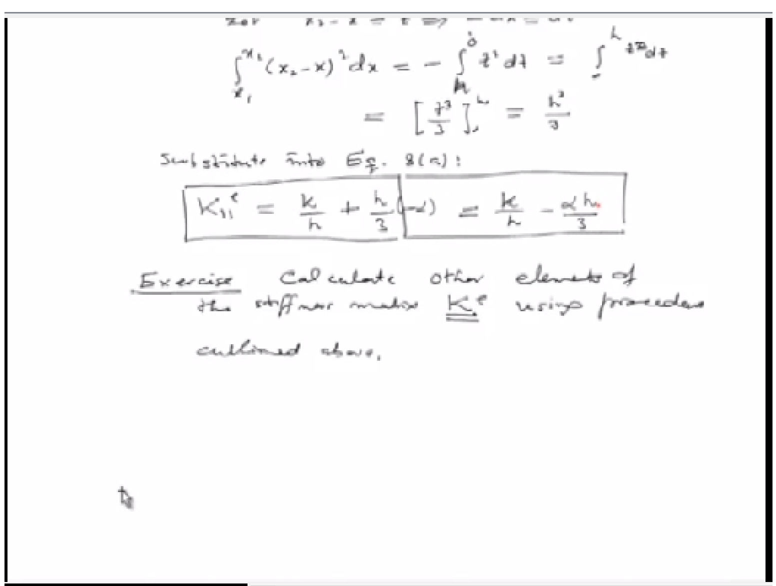


So K11e, this will be equal to integral of K times dN1/dx*dN1/dx-alpha*N1N1*dx. Now there are 2 parts or 2 terms in this integral. So let us evaluate them separately. So the first part is integral x1 to x2, dN1 that was equal to -1/x. So -1/h*-1/hdx, so it simply gives us h/h squared or this gives us value equal to 1/h.

Similarly our integral N1N1dx, this is integral of x2-x whole square dx x1 to x2 and if you make this appropriate substitutions, so this will become 0 to h, if you substitute x2-x=t, so let us do this substitution, so let x2-x=t, this tells us that -dx=dt. So our integral x1 to x2 x2-x whole square dx=-h to 0 t square dt which can also be written as 0 to h t cube dt. So this gives us t cube/3 0 to h or in other word from here we will get h cube/3.

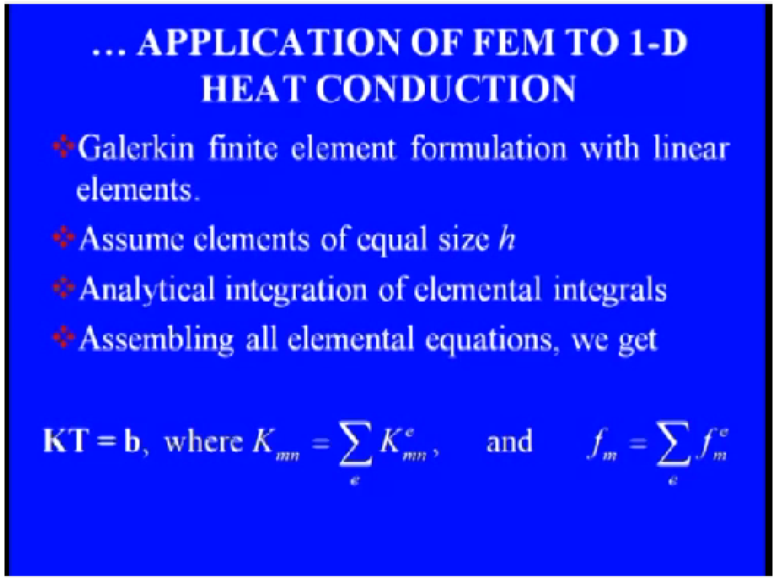
Okay, so now we are ready to substitute the values in our, let us call it equation 8a, so substitute into equation 8a and we will get K11e=k/h+h/e. So we have evaluated 1 term. You can repeat exactly the same procedure to evaluate the other terms K12e K22e and so on, okay. So if you apply this process of integration which I would leave as a simple exercise to you.

(Refer Slide Time: 46:04)



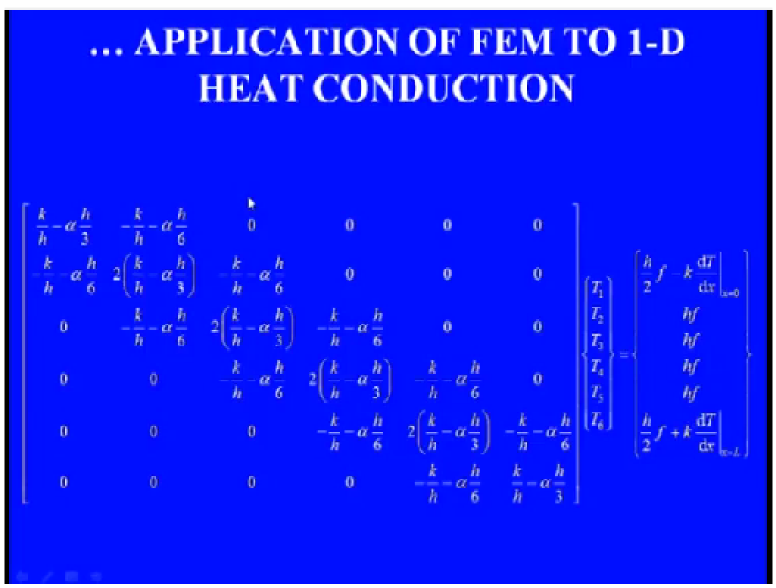
So your exercise would be in calculate other elements of the stiffness matrix Ke using procedure outlined above. So if you repeat that exercise, now I am going to sum it up that what we will get after putting all these terms.

(Refer Slide Time: 46:49)



Once we have got all the elemental integrals Kmne, we can sum them up and find out what would be our global stiffness matrix Kmn and similarly we can also find out what will be our global load vector f.

(Refer Slide Time: 47:09)



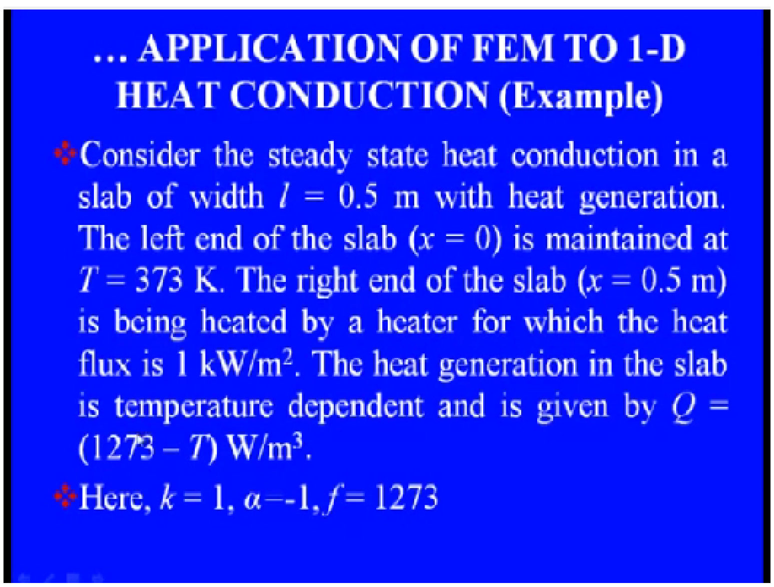
So that is how the things look like for our, let us say we have used a subdivision into 5 of our domain 0 to L shows how the terms look like, k/h-alpha*h/3. So in our elemental level equations what we had, this h/3 was multiplied by factor -alpha. So this becomes k/h-alpha h/3, okay. So that is how our things look like. Again a tridiagonal matrix, okay, so first row is k/h-alpha*h/3, this is our K11. K12 is -k/h-alpha*h/6 and so on.

So you can easily calculate all these entries multiplied by T1, T2, T3 to T6 and this is what we get on our right-hand side, h/2f-kdT/dx at 0. This is the term which we will get for the first node or rather when we consider the first element, the contribution from the natural boundary condition, there comes in picture and similarly, the contribution at the right end. Remaining terms is just simply coming from the source term and this equation has to be solved for after application of the boundary conditions.

For instance, if, let us say temperature was specified at the first node T1. So in that case, what we will do. This particular diagonal element would be set to 1 and rest would be set to 0 in the first row and the right-hand side term, this b1 would be set to T1 of the specified value, let us call it Ta. If the flux boundary condition was specified, substitute for the flux value here and then modify the load vector.

If convective boundary conditions were specified, there will be some contribution coming from here and that has to be added to the diagonal elements of the last row. So that is how we would apply your boundary conditions.

(Refer Slide Time: 49:33)

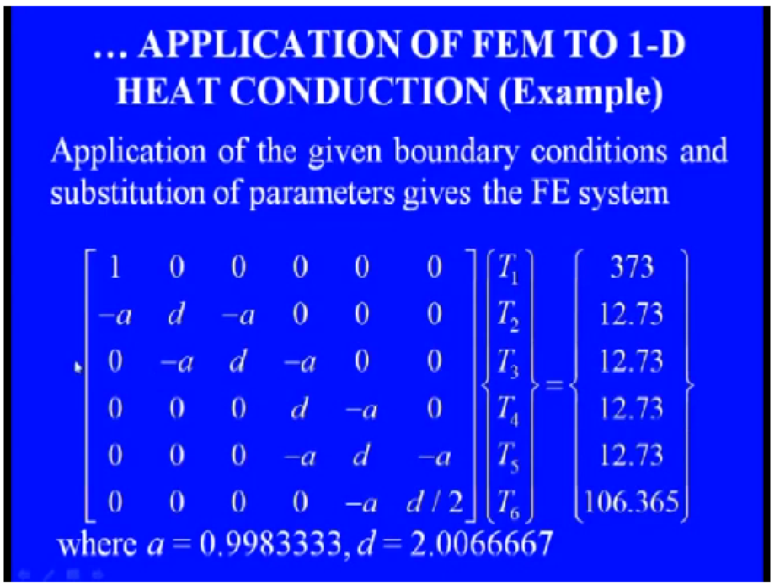


Now let us have look at one example. So consider our continuing example which we have considered earlier in the case of finite difference and finite volume method. Steady state heat conduction in a slab of width l=0.5 m with heat generation. The left end is maintained at T=373.

Right end is being heated by heater with a heat flux of 1 kilowatt per meter square and heat generation is given by Q=1273-T w cube.

So if you compare with our formal notation, what we have seen, conductivity k=1, the linear coefficient which we had in for the source term alpha=-1 and the constant term, the source description f=1273. So if you substitute these terms in our equations we derived earlier and we will modify, we will multiply the things a little bit.

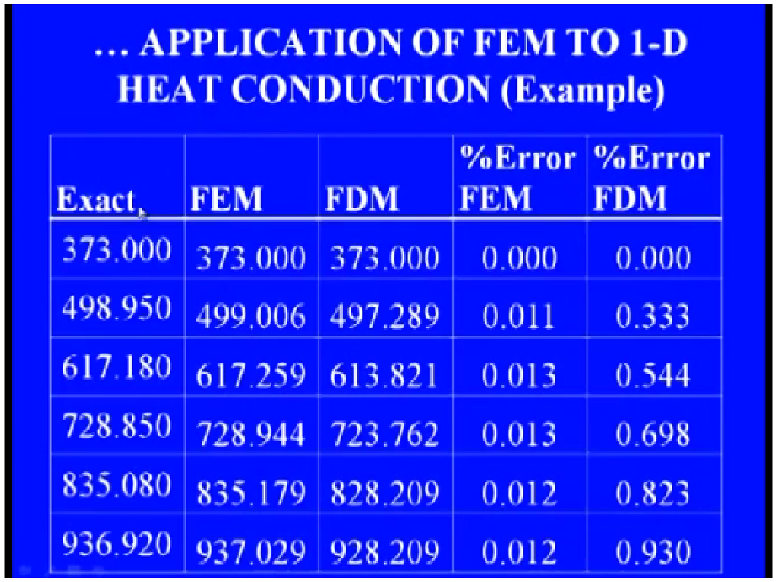
(Refer Slide Time: 50:28)



Let us say, remaining equations have been normalised a bit. First equation, you can clearly see, temperature is specified T1=373. So we have set the diagonal, this main diagonal term as 1. Remaining terms in this row as 0 and the right-hand side becomes 373 and so on, okay. Remaining terms that have been normalised, so that they appear close to the ordered 1, that is why we get this d is 2.00667 and of diagonal terms a take the value 0.9983333 and so on.

So this is our tridiagonal system which we got because we have chosen a linear element and can be solved using our TDM algorithm and this is a comparison of the solution.

(Refer Slide Time: 51:24)

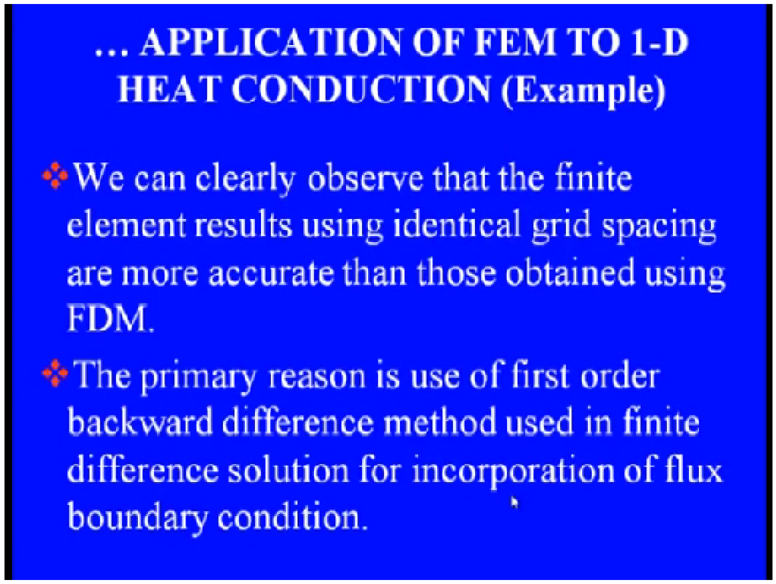

… APPLICATION OF FEM TO 1-D HEAT CONDUCTION (Example)

Exact	FEM	FDM	%Error FEM	%Error FDM
373.000	373.000	373.000	0.000	0.000
498.950	499.006	497.289	0.011	0.333
617.180	617.259	613.821	0.013	0.544
728.850	728.944	723.762	0.013	0.698
835.080	835.179	828.209	0.012	0.823
936.920	937.029	928.209	0.012	0.930

So exact solution at different nodes, of course at node 1, temperature was specified, so FEM and FDM, they will give the exact solution we have incorporated that exactly but you see the percentage errors here. The percentage errors with FECM are less than 0.01% everywhere, whereas those appreciable error in finite difference formulation and the part of reason was simple enough that though we have used the same grid size in finite element as well as finite difference formulation.

In finite differences, the flux boundary condition was incorporated using what we call a first order accurate backward difference scale. So that leads to the spoiling of this finite differential solution. The overall accuracy becomes a first-order whereas in finite elements, the accuracy is of the second order. So finite element solutions you can also compare them with the solutions which we have discussed earlier of the same problem using finite volume. The 2 solutions are fairly same, okay.

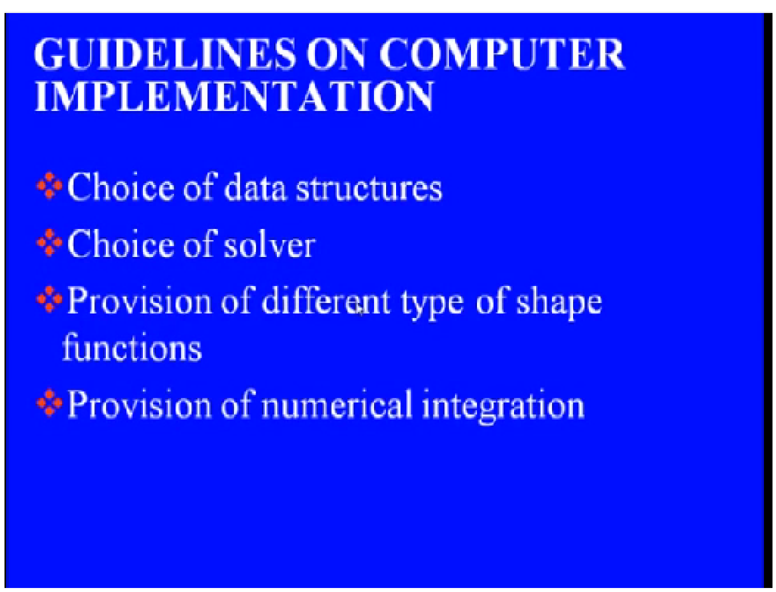
(Refer Slide Time: 52:29)



So that is what we have observed here that the finite element results using identical grid spacing are more accurate than those obtained using FEM and we have outlined the reason, because when a difference formulation becomes a first-order due to the incorporation of flux boundary condition. In finite element, the flux boundary condition was incorporated directly or and in what we call a natural fashion, so there was no approximation involved in the incorporation of the flux boundary conditions.

So that was the primary reason for the better accuracy of finite element method. Now a brief look at some guidelines for the computer implementation.

(Refer Slide Time: 53:08)



Now whenever you want to write a finite element code, you will have to first make your choice of data structures in the same way as we did for finite difference of the finite element. There is slightly 1 difference here, in the case of a structure finite difference or finite volume, we knew that we will get what we call multidiagonal structure of the matrix. So we could easily choose those number of diagonals and store them as 1-dimensional array.

Now in case of finite elements, if you move over to unstructured grid in the 2-dimension or 3-dimensions, the things will change, the type data structure that would depend on the choice of our solver, whether we want to use an element by element solver or we want to use general-purpose solvers like iterative solve like PCG or GMRES and so on and what way we will store our elemental matrix. So all these things will decide our choice of the data structure in the design of a finite element code.

Now once of course you have decided in the data structures, the code you must also provide for the provision of different types of shape functions. For instance, for 1-dimensional heat conduction, I would like you to write a simple finite element program. First use it or write a code which makes use of leader interpolation functions and write one which makes use of quadratic interpolation functions. In the case of 1-dimensions though, the things are very simple.

We can evaluate the integrals analytically but what I would encourage is that you put the provision for the shape functions and go for Gaussian quadrature and compare the 2 results. The results which we would obtain if we make use of analytical integration and the one which we would obtain from numerical integration. So provide a provision for the numerical integration in your finite element code.

(Refer Slide Time: 55:03)

So these words, I would like to put a stop to our discussion on finite element method. The method is very versatile, very capable but also you have to learn it, learn many more things before you can write a multidimensional finite element code and for further reading, please refer to the books by Zienkiewicz, Taylor and Zhu or the book by Reddy.

Some details are also available in Computation Fluid Dynamics book by Chung but if you looking for the complete solution strategy, it is better to go and have a look at Zienkiewicz' and Taylor's book, the finite element method, its basis in fundamentals which also provides you some discussions on the data structures and the storage aspects of finite element matrices.